Solving Asymmetric Travelling Salesman Problem Using Group Constructive Crossover

W. K. Alrashdan[1], S. Abu_owida and W. Alsharafat[2]
Al- Balqa' Applied University, Jordan, Irbid[1]
Prince Hussein Bin Abdullah Faculty of Information Technology Al al-Bayt University, Mafraq, Jordan[2]

Summary
Traveling salesman Problem (TSP) is one of the important problems which classified as an NP-complete problem. Thus, this paper suggested an enhancement on Genetic Algorithm (GA) through the crossover process and its probability to optimize Asymmetric TSP (ATSP). The proposed crossover modification called Group Constructive Crossover (GCX) which used to find an optimal solution. The efficiency of the GCX is conducted by comparing it with some of existing crossover methods such as a Sequential Constructive crossover (SCX) and the ordered clustered TSP which called A Hybrid Genetic Algorithm (HGA). All comparisons were conducted by using dataset benchmark called TSPLIB.

Key words:
Genetic Algorithm, Crossover, GCX, ATSP.

1. Introduction

The TSP is an old problem and difficult problem that is hard to solve because it is one of an NP-complete problem [1]. TSP can be defined as a graph with vertex (cities), and edge (distance, or travel time etc.) [1,2]. The TSP implemented to find the shortest tour where every city has been visited only once and then to return to the start city [1,2]. The TSP is divided into two basic types: Symmetric TSP (STSP) where the distances between two cities are equivalent in both sides; forward and backward with(n −1)!/2 possible solutions. In opposite, the Asymmetric TSP (ATSP) where the distance between two cities don't equal forward and backward with (n-1)! possible solutions [2]. ATSP is a complete direct graph which represented as G= (V, E) where V= {1,…., n} is a city set, E={ (i,j): i,j€V, i<j } is an edge set and {(i,j): i,j €V, i≠j is an arc set [2]. A cost matrix C=(cii) is defined on E or on A. The cost matrix satisfies the triangle inequality whenever cij≤ Cik + C kj, for all i,j,k[1,2],

2. Genetic Algorithm(GA)

Heuristic searching algorithms as GA depends on the evolutionary ideas of natural selection where GA implemented in the intelligent exploitation of a random search solution in a specified search space for finding optimal solutions [3].
GA uses the idea of the survival of the fittest and produces population to create the innovative search strategy. At the beginning, GA consists of a set of chromosomes, individuals, called population, which symbolize the possible solution for a stated problem. Then GA creates new chromosomes from previous population, which called a generation where this mission accomplished by applying crossover and mutation on populations [4].

3. Literature Review

Different heuristic algorithms have been enhanced and modified to find an optimization solution for ATSP. The work in this paper is inspired and guided by previous research concerning in ATSP environment.

A. Hybrid GA: refers to the algorithm of combining GA with other approaches avoid sticking in local search, for finding a better solution in terms of either the quality of the solution or the computation time. The combination of GA and local search is favorable for finding near optimal solution [5]. Researchers in [5] have exploited Simulated Annealing (SA) algorithm to solve the TSP by proposing a new Hybrid Genetic and Simulated Annealing Algorithm (HGSAA). The HGSAA used two individuals as a population and applied multi-crossover. The crossover containing ahead, middle and the tail after that restored arranged to produce different offspring. The HGSAA used inversion mutation operation, where random sub tour was selected from the second individual then was inverted. The algorithm was stuck after a number of generations, so the hybrid algorithm changed it's population which allowed uphill jumped to a higher-cost solution in order to avoid getting trapped in local search and to get a better solution. The experiment showed that the HGSAA algorithm provided good results within a reasonable time.

1. Combine GA with the heuristic method: heuristic techniques have been used to find the efficient solution [6]. Combined the local search used to find the optimal solution locally by applying the Lin - Kernighan algorithm

Manuscript received September 5, 2017
Manuscript revised September 20, 2017

for local optimization. In addition, GA used to find the optimal solutions globally. Lin-Kernighan algorithm concerned about exchange couple of sub tours to become a new tour. It was a generalization algorithm which works on two-points and Three-points by switching two or three paths to reduce the tour distance; to be shorter than before. Lin-Kernighan was adapted in each step to determine how many possible paths between cities to be switched to find a shorter tour. The method used STSP datasets and ATSP datasets, the method produced a high quality solution in reasonable time. In [7], researchers have suggested a method to research the optimal solution; which was the shortest route for traversing between cities. In this method, the solution was generated without having any prior knowledge about the problem. This method works by selecting four parents and applying crossover and mutation to each parent and then select the best two parents after applying crossover and mutation. The proposed method was characterized as a flexible, easier and faster method.

2. Combine GA with Neural Network: Combining neural network with GA leads to train neural nets to choose their structure aspects like the functions of their neurons [8]. In [8], Mahajan and Gaganpreet have offered a solution by implementation of GA to reach the maximal approximation of TSP in term of cost reduction.

B. Clustering GA: is an important technique for reducing the solution values of large problems space. Researchers have proposed the ordered clustered TSP, which called (HGA). The HGA works by dividing a dataset into defined clusters which considered the main difference between HGA and traditional TSP. The goal of the HGA was to reach an optimal solution at minimal cost.

C. Enhancing crossover operator: in GA, the crossover is the most important operations which lead the researchers in [1] to focus on suggesting and enhancing crossover for solving TSP problem. In [1], researcher has suggested Sequential Constructive Crossover (SCX). The algorithm divided into two stages, the first stage works as follows; if the selected child was found in the parents, in this case, we would choose the least cost in the parents. Else, if the child was selected were not present in the parents then choose the city according to squeal number. In [11], the proposed Shared Crossover (SX) applied to find near-optimal solutions for TSP problem. The SX used and shared neighbors to ensure that the closest cities have the highest chance to be transported to the next generation. If the node has no shared neighbor between the two parent, chromosomes, or the shared neighbor, then SX will select the nearest neighbor to the node. The main advantage of SX against other algorithms is reducing the execution time and population numbers.

4. Research Methodology

We have a proposed a crossover operator called Group Constructive Crossover (GCX) to reach the best solution of ATSP. Where GCX works as appeared in Figure 1.

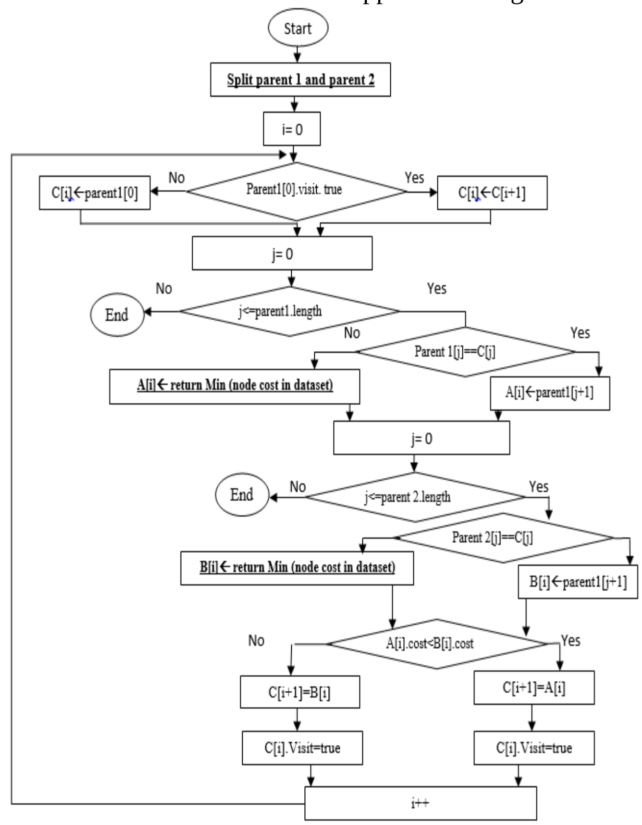

Fig. 1 GCX flow chart

We will clarify GCX by giving an example, as in Figure 2. The chromosomes were randomly selected as follows :

- Chromosome 1: 5-8-6-9-4-2-1-3-7=186.
- Chromosome 2: 8-6-4-1-2-3-9-7-5=179.

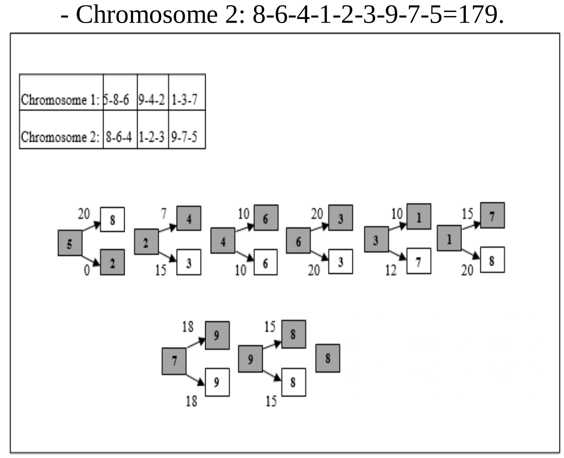

Fig. 2 Tours of GCX

As shown in Figure 2, the GCX splits chromosome 1 and chromosome 2 to group each group has 3 genes, after that select node 5 as an initial city. The next node after node 5 in chromosome 1 is node 8, but doesn't exist on chromosome 2 in this case, here GCX different from SCX, in GCX we select the minimum cost in the cost matrix as given in table1, that is node 2. Since c58 = 20 > 0 = c52, we select node 2 as the next node. The next node after node 2 in chromosome 1 doesn't exist in this case select the minimum cost in the cost matrix as given in table1, that is node 4, but exist on chromosome 2 that is node 3, since c24= 7<15= c23 we select node 4 as the next node. The next node after node 4 in chromosome 1 and chromosome 2 doesn't exist, in this case we select the minimum cost in the cost matrix as given in table1, that is node 6. The next node after node 6 in chromosome 1 and chromosome 2 don't exist, in this case we select the minimum cost in the cost matrix as given in table1, that is node 3. The next node after node 3 in chromosome 1 is node 7, but doesn't exist on chromosome 2, in this case, we select the minimum cost, that is node 1. Since c37 = 12 > 10 = c31, we select node 1 as the next node.

The next node after node 1 in chromosome 1 is node 7, but doesn't exist on chromosome 2 in this case, select the minimum cost in the cost matrix as given in table1, that is node 8. Since c17 = 15 < 20 = c18, we select node 7 as the next node. The next node after node 7 in chromosome 1 and chromosome 2 don't exist, in this case we select the minimum cost, that is node 9. The next node after node 9 in chromosome 1 and chromosome 2 don't exist, in this case we select the minimum cost in the cost matrix as given in table1 that is node 8. The total cost of the GCX is 130.

Table 1: The Cost Matrix

	1	2	3	4	5	6	7	8	9
1	999	10	30	25	31	30	15	35	30
2	18	999	15	7	48	18	28	36	25
3	10	2	999	0	12	48	12	24	22
4	25	15	20	999	25	10	15	31	21
5	22	0	1	32	999	26	10	20	40
6	28	30	20	28	41	999	30	31	30
7	30	21	12	15	20	15	999	27	18
8	10	15	26	11	21	11	20	999	31
9	10	15	26	20	40	30	28	15	999

5. Experimental Results

We will present experimental results and evaluation of the GCX using error of the solution for each dataset and average. The error is measured by [12] :

Error % = ((| Best solution - optimal solution|) / (optimal solution)) x 100

To provide a good judgment on the GCX, we compare GCX results with different researchers result whose mainly adapted GA in their work to solve ATSP and those researches are:

1. Sequential Constructive Crossover Operator [1].
2. The Ordered Clustered TSP: A Hybrid Genetic Algorithm (HGA) [10].

To perform the tests, the benchmark TSPLIB was used for the ATSP tests. The dataset chosen for our test is ftv33, ftv38, ftv47, Ftv55 and Ftv64. In GA, different parameters value must be specified earlier when conducting experiments to find optimal or near optimal solutions. These parameters as follows:

- Population Size: 200
- Selection parent: ranking
- Crossover: GCX
- Crossover probability: 0.9
- Mutation probability: 0.09
- Termination condition: 2000

Table 2 will Summary the result of different crossover operator.

Table 2: Crossover Operator Results

datasets	SCX		HGA		GCX	
	Error %	AVG. error %	Error %	AVG. error %	Error %	Avg. error %
Ftv33	0.00	3.58	0.05	0.06	0.03	0.05
Ftv38	0.24	0.46	0.03	0.04	0.01	0.03
Ftv47	0.51	1.73	0.10	0.13	0.08	0.10
Ftv55	0.62	1.45	0.10	0.11	0.11	0.12
Ftv64	0.49	1.54	0.09	0.11	0.07	0.09

As shown in Table 2, GCX achieved an improvement in error rate compared with SCX and HGA in five datasets; Ftv33, Ftv38, Ftv47, Ftv55 and Ftv64 datasets. In addition, the average path cost for GCX achieved best result in five datasets compared with SCX. In addition, the average of error of GCX gain better results compared with HGA.

According to Figure 3 and Figure 4, we noticed the following:

1. GCX achieved an improvement in the average of error compared with SCX and HGA.
2. GCX gain less error in the majority of datasets, at least 4 datasets from 5 datasets.

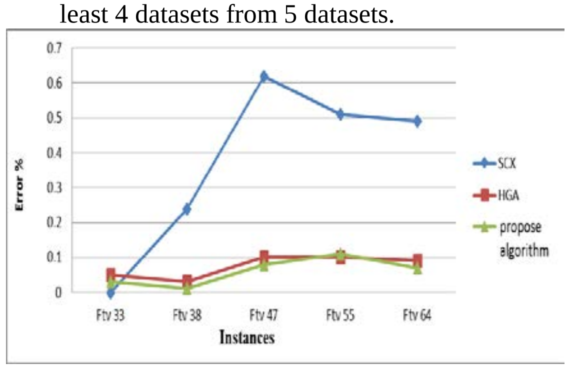

Fig. 3 Error solutions as a % of TSPLIB

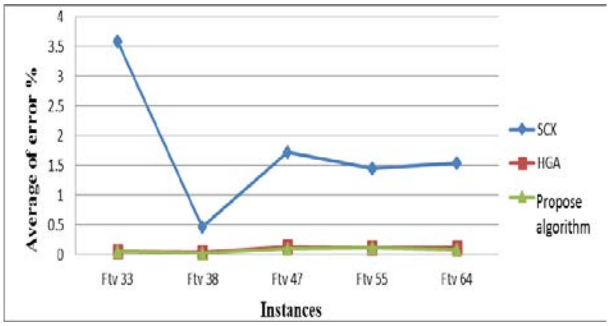

Fig. 4 Average of error as % of TSPLIB

6. Conclusion

The main achievement of this paper concerns about providing an enhanced crossover operation using GA with their probabilities, which called Group Constructive Crossover (GCX). The Experimental results demonstrate the successful properties of GCX on a TSPLB dataset in the range 17-64 cities. GCX has been examined against a Sequential Constructive crossover (SCX) [1], and ordered clustered TSP: A Hybrid Genetic Algorithm (HGA) [5] where the experiments show that GCX achieved an efficient result in producing a high quality solution for the benchmark datasets. Many improving issues could be considered as future work, some of them concerns about combining GA with hill climbing methods to avoid stuck in local search. Also, expand comparisons on large dataset.

References

[1] Zakir H. Ahmed, "Genetic algorithm for the traveling salesman problem using sequential constructive crossover operator ", International Journal of Biometrics & Bioinformatics, Volume 3 (6), 96 -105, 2010.
[2] Matai.R, "Traveling salesman problem: an overview of applications, formulations, and solution approaches. Traveling salesman problem, theory and applications", 1-24, (2011).
[3] Goldberg. D, "Genetic algorithms in search, optimization and machine learning", (1st ed), New York: Addison-Wesley, 1989.
[4]
[5] Mitchell.M, "An introduction to genetic algorithms", (5th ed), London: A Bradford Book the MIT Press(1999).
[6]
[7] Elhaddad .Y and Sallabi. O," A new hybrid genetic and simulated annealing algorithm to solve the traveling salesman problem", proceedings of the world congress on engineering, Volume 1, London, 10 -15, (2010).
[8]
[9] Freisleben. B and Mer. P, "A genetic local search algorithm for solving symmetric and asymmetric traveling salesman problems", IEEE International Conference on Evolutionary Computation, Nagoya 616 – 621, (1996).
[10] Asim. M., Gopalia. R. and Swar. S, "Traveling salesman problem using genetic algorithm", International Journal of Latest Trends in Engineering and Technology, Volume 3 (3), (2010).
[11] Mahajan. R, Gaganpreet. K, "Neural networks using genetic algorithms", in international journal of computer applications, 2013.
[12] Zakir H. Ahmed, "The ordered clustered travelling salesman problem: a hybrid genetic algorithm", the Scientific World Journal, Volume 2014, 1-13, 2014.
[13] Al kassbeh. M., Alabadleh. A and Al-Ramadeen. T,"Shared Crossover Method for Solving Traveling Salesman Problem", International Journal of Basic and Applied Sciences, Volume 1(6), 153 -158, 2012.
[14] TSPLIB (Online) Available at: http://www.iwr.uni-heidelberg.de/iwr/comopt/software/TSPLIB95/.